BOLT BERANEK AND NEWMAN INC

CONSULTING · DEVELOPMENT · RESEARCH

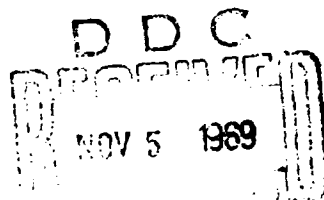Report No. 1890   October 1969

AD696122

INTERFACE MESSAGE PROCESSORS FOR
THE ARPA COMPUTER NETWORK

QUARTERLY TECHNICAL REPORT NO. 3
1 July 1969 to 30 September 1969

Principal Investigator: Mr. Frank E. Heart
                          Telephone (617) 491-1850, Ext. 470

D D C
NOV 5 1969

CLEARINGHOUSE
for Federal Scientific & Technical
Information Springfield Va. 22151

CAMBRIDGE   NEW YORK   CHICAGO   LOS ANGELES

16

Report No. 1890   October 1969

INTERFACE MESSAGE PROCESSORS FOR
THE ARPA COMPUTER NETWORK

QUARTERLY TECHNICAL REPORT NO. 3
1 July 1969 to 30 September 1969

Principal Investigator: Mr. Frank E. Heart
                          Telephone (617) 491-1850, Ext. 470

Report No. 1890   Bolt Beranek and Newman Inc.

TABLE OF CONTENTS

1. INTRODUCTION

This Quarterly Technical Report No. 3 describes several aspects of our progress on the ARPA computer network during the third quarter of 1969. During this period, the first IMP was delivered to UCLA on schedule with an operational program. The IMP successfully communicated with the UCLA Host computer (a Sigma 7).

In Section 2, we describe the test programs developed, the testing procedure used, and the technical problems encountered in installing the initial IMP. In Section 3, we outline several new features that have been incorporated into the operational IMP program.described in our Quarterly Technical Report No. 2 (BBN Report No. 1837). We will soon make these features available in a second version of the operational program. We have begun a preliminary study of the IMP program in an attempt to understand its performance. A few projected measures of program performance are presented in Section 4.

Documentation during this quarter consisted of two minor revisions of the Host specification (BBN Report No. 1822).

2.   HARDWARE CHECKOUT AND INSTALLATION

During this quarter, we tested the IMPs intensively. After being tested separately, the IMPs were combined into small networks of two or three locally connected IMPs and then retested. Upon installation at UCLA, the IMP was tested further.

A.   Test Program Development

An extensive program has been developed for checkout and testing of the IMP. The program consists of two parts: first, a section that performs one-time tests on several special IMP features (watchdog timer, automatic restart, memory protection, power-fail interrupt, etc.); and second, a loop that repeatedly drives a selectable data pattern through the interfaces to compare incoming data with outgoing data for errors. The data can be driven through a crosspatched interface, through a locally looped modem, through a phone line looped at a remote location, or to another IMP performing an identical test.

The tests have uncovered bad cables and logic packs, a number of wiring deficiencies, two minor interace design errors, and a design problem in the DDP-516 data-channel hardware. We are preparing a manual that describes verification, test, and installation procedures and discusses the test program in detail.

B.   Test Cell Activity

During this quarter, Honeywell delivered IMPs Number 2 and 3. As with IMP No. 1, these machines contained a considerable number of faults which were debugged and corrected in the Test Cell.

A small temporary hardware patch to the Modem interfaces in IMPs 1 and 2 made possible the direct connection of these

interfaces, thereby removing the need for intervening modems.
This modification allowed for the construction of a three node
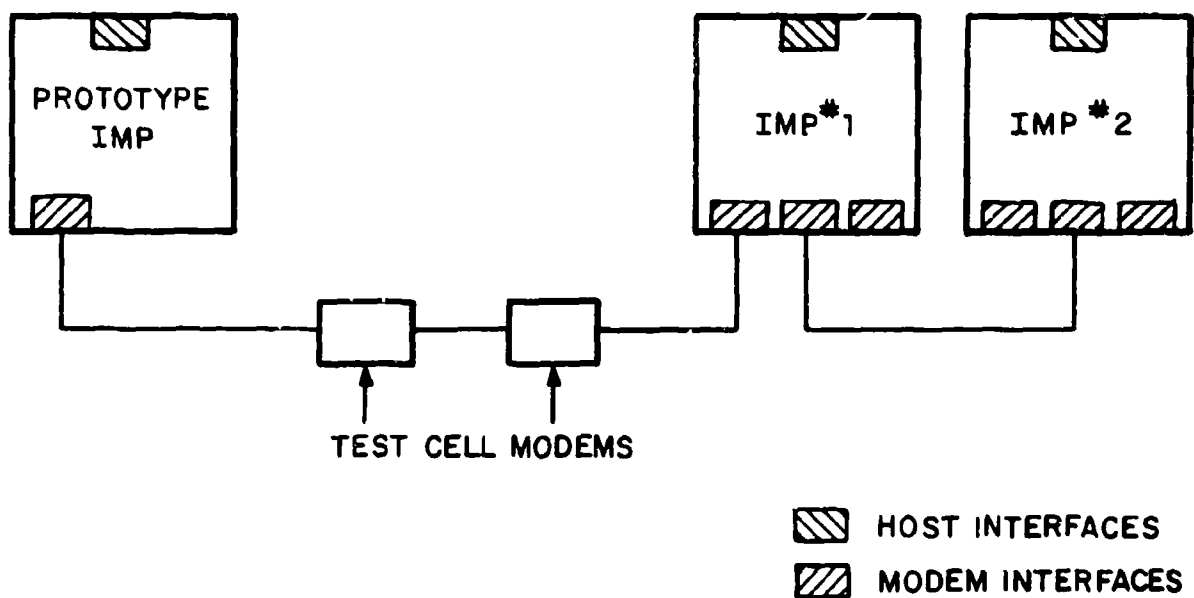pseudo-network as shown in Figure 1 below:

FIG. 1

This configuration provides a reasonably sophisticated
setting for hardware testing and also allows for debugging of
configuration dependent features (e.g., routing) of the
operational program. An equivalent configuration was constructed
using IMP No. 3 to replace IMP No. 1 after the latter had been
shipped to UCLA.

C.   Initial Installation Activity

Within a few days of its delivery to UCLA on Saturday, August 30, the IMP was connected to and operating with both the Sigma 7 and the phone company equipment. Intensive testing revealed a minor design error in the IMP's standard Host interface and a minor design error and some bad components in the Host's special interface. After these errors were corrected, tests were conducted with the UCLA-SRI phone line looped at the SRI end, and messages were successfully sent around this loop. During installation, we decided that we would like to test the phone company equipment but we found that the program used in the BBN Test Cell was not appropriate for studying phone-line error characteristics. We are presently writing a program for this purpose.

At the time of installation, we recognized a need for the Sigma 7 Host to have its own test program for communication with its IMP. A cooperative UCLA-BBN effort resulted in a simple program to send and receive character strings between the IMP and Host teletypes. This approach proved so successful that we plan to encourage the use of similar programs in all future installations.

We found the phone company installations at UCLA and SRI to be inconsistent with regard to physical configurations of voice circuits cabinetry, original design, etc. These difficulties were reported to the telephone company.

3. SOFTWARE DEVELOPMENT

The software effort during this quarter was devoted to implementing the program design described in our Quarterly Technical Report No. 2. Version 1 of the operational program was delivered to UCLA with the first IMP. At this time, implementation is almost complete.

During this implementation period, new software features involving error-recovery procedures have been added to the program. These procedures handle the failure of an IMP or a Host, with consequent loss of whole or partial messages from the network. We feel that after a reasonable period, on the order of many minutes, all trace of such an event should be eliminated from the network and that the user should be informed of the occurrence.

Error-recovery procedures fall into two categories: the response of the network to an IMP or a Host failure and the response of an IMP to its own failure.

A. Network Failure Recovery

An IMP may detect a network failure in one of three ways:
1. A packet expected for reassembly of a multiple packet message never arrives.
2. A link in the link table of the transmit IMP is never unblocked.
3. The Host does not take a message from its IMP.

If a message is not fully reassembled in 15 minutes, the system presumes a failure. The message is discarded and a RFNM returned with a "transmission incomplete" bit set. This

RFNM, in turn, is passed along to the transmitting Host as
Error Message Number 9.

If the Host has not taken a message after 15 minutes, the
system presumes that it will never take the message. Therefore,
as in the previous case, the message is discarded and a RFNM
with the incomplete transmission bit is returned to the source
Host.

If a link remains blocked for longer than 20 minutes, the
system again presumes a failure, perhaps a lost RFNM or a lost
message. The link is unblocked and an incomplete transmission
error message is sent to the source Host. The delay is slightly
longer for this failure so that the other failure mechanisms
will have a chance to operate and unblock the link.

All three failures involve an event that takes much longer
that it should. For the present, we have tried to pick reason-
able time limits for each case; as we discover more about the
behavior of the network, we will be able to define these limits
more exactly.

In all three cases, Error Message No. 9 is given to the
transmitting Host. We expect that failures of this sort will
be infrequent enough to permit the human operator controlling
the Host transmission to determine how to proceed.

B.   IMP Failure Recovery

An IMP can recover from its own failure in two ways. In
the event of power failure, a hardware feature permits the IMP
to turn off the program before the program destroys itself.
When power returns, the IMP restarts automatically. We con-
sidered several possibilities for handling the packets found

in an IMP during a power failure and concluded that no plan to
salvage the packets was both practical and foolproof. For
example, we cannot know whether the packet in transmission at
the time of failure successfully left the machine before the
power failed. Should that packet be reintroduced into the
network after a lengthy delay, it might actually be delivered
twice! Therefore, we decided simply to discard all the packets
and restart the IMP program.

The second recovery mechanism is the "watchdog timer",
which transfers control to protected memory whenever the pro-
gram neglects this timer for about one minute. Everything
unique to a particular IMP must reside in its protected memory.
Only one register (containing the IMP number) currently differs
from IMP to IMP.

We presume that the program in unprotected memory is
destroyed either through a hardware transient or software fail-
ure. The program in protected memory sends a reload request
down a phone line selected at random. The neighboring IMP re-
sponds by sending a copy of its whole program back on that phone
line. A normal IMP would discard this message because it is too
long, but the IMP in trouble can reload its program. The pro-
cess of reloading from the network takes only a few seconds and
can be repeated until successful. This feature of loading from
the network would permit delivery and incorporation of a new
version of software through the network. However, we still view
paper tape as the primary input medium.

C. Stopping an IMP

Care must be taken to stop a working IMP without introducing
network failures. Therefore, we have implemented a "clean stop"

feature (a special switch) that shuts down the IMP without los-
ing messages. The program initiates the following sequence of
events when the IMP is taken down cleanly:

1.   Sends the Host an "IMP going down" message.
2.   Waits 5 seconds to let the Host finish network
     transactions.
3.   Refuses messages from the Host and notifies the
     network that the Host is dead.
4.   Waits 5 seconds to let other Hosts learn that
     this Host is dead.
5.   Refuses messages from the network.
6.   Waits 5 seconds to allow its IMP to empty of
     store and forward messages.
7.   Stops.

4.   PROJECTED IMP PERFORMANCE

During the last quarter we began to study the projected performance of the IMP. This study was based upon a recent version of the operational program and provides only pre-liminary data. The IMP has not yet been tested under heavy load conditions and consequently no experimental data is available. In the following paragraphs, we present a few conclusions about IMP performance.

A.   Capacity in Connected Lines

The amount of traffic flowing on a fifty kilobit line fully loaded with store and forward packets is adopted as a unit traffic load on the IMP. We call this unit an *effective channel*. Thus, a fifty kilobit line offers at most one ef-fective-channel load, while a 230.4 kilobit line offers at most a load of 4.6 effective channels. Conveniently, the processing time for a message on the Host line is about equal to the processing time for the same message on a phone line; thus, Host lines and phone lines are equal with regard to effective-channel traffic.

The computational capacity of the IMP is a function of message length. For a load consisting only of short messages (one word), the capacity is seven effective channels. For the longest messages (eight packets), the capacity is nineteen effective channels.

B.   IMP Throughput

We adopt the IMP throughput in bits/second as a measure of IMP performance. The throughput is the maximum number of

Host data bits that may traverse an IMP each second. The
actual number of bits entering the IMP each second is some-
what larger than the througput because of such message
overhead as headers, RFNMs and acknowledgements. Each packet
on the phone lines contains seventeen characters of overhead,
thirteen of which are removed before the packet enters an
IMP.

The maximum IMP throughput of approximately 700,000
bits/second is achieved with large (8 packet) messages on
nineteen effective channels. A curve of maximum throughput
as a function of message length is shown in Figure 2. The
difference between the throughput curve and the line traffic
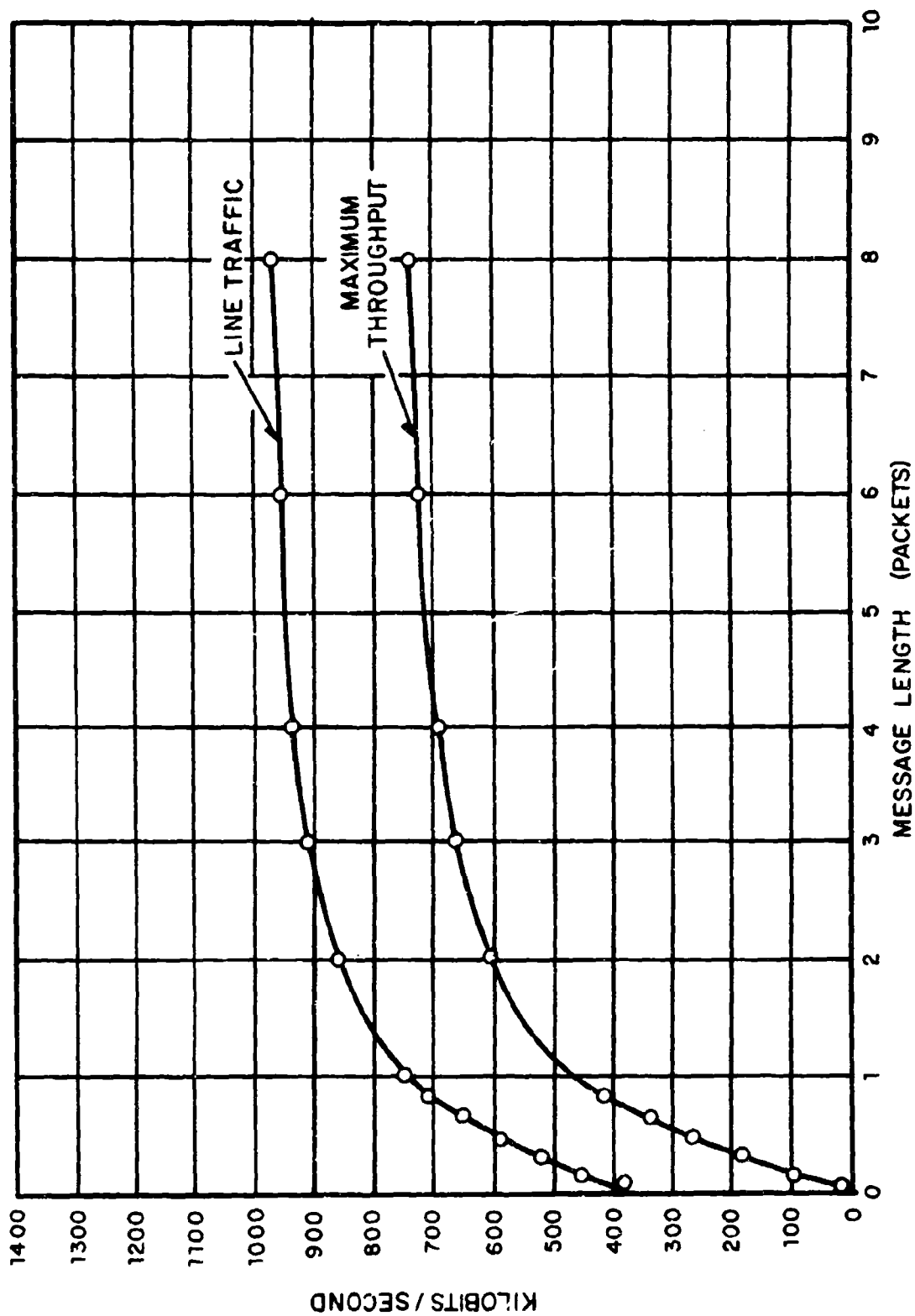curve represents overhead.

FIG. 2

MESSAGE LENGTH (PACKETS)

KILOBITS / SECOND

LINE TRAFFIC

MAXIMUM THROUGHPUT

DOCUMENT CONTROL DATA · R & D

(Security classification of title, body of abstract and indexing annotation must be entered when the overall report is classified)

1. ORIGINATING ACTIVITY *(Corporate author)*	2a. REPORT SECURITY CLASSIFICATION
Bolt Beranek and Newman Inc. 50 Moulton Street Cambridge, Massachusetts 02138	UNCLASSIFIED
	2b. GROUP

3. REPORT TITLE

QUARTERLY TECHNICAL REPORT No. 3
1 July 1969 to 30 September 1969

4. DESCRIPTIVE NOTES *(Type of report and inclusive dates)*

5. AUTHOR(S) *(First name, middle initial, last name)*

Bolt Beranek and Newman Inc.

6. REPORT DATE	7a. TOTAL NO. OF PAGES	7b. NO. OF REFS
October 1969	13	

8a. CONTRACT OR GRANT NO.	9a. ORIGINATOR'S REPORT NUMBER(S)
DAHC 15-69-C-0179	
b. PROJECT NO.	BBN Report No. 1890
c.	9b. OTHER REPORT NO(S) *(Any other numbers that may be assigned this report)*
d.	

10. DISTRIBUTION STATEMENT

11. SUPPLEMENTARY NOTES	12. SPONSORING MILITARY ACTIVITY
	Advanced Research Projects Agency Washington, D.C. 20301

13. ABSTRACT

The basic function of the IMP computer network is to allow large existing time-shared (Host) computers with different system configurations to communicate with each other. Each IMP (Interface Message Processor) computer accepts messages for its Host from other Host computers and transmits messages from its Host to other Hosts. Since there will not always be a direct link between two Hosts that wish to communicate, individual IMPs will, from time to time, perform the function of transferring a message between Hosts that are not directly connected. This then leads to the two basic IMP configurations — interfacing between Host computers and acting as a message switcher in the IMP network. The message switching is performed as a store and forward operation. Each IMP adapts its message routine to the condition of those portions of the IMP network to which it is connected. IMPs regularly measure network performance and report in special messages to the network measurement center. Provision of a tracing capability permits the net operation to be studied comprehensively. An automatic trouble reporting capability detects a variety of network difficulties and reports them to an interested Host. An IMP can throw away packets that it has received but not yet acknowledged, transmitting packets to other IMPs at its own discretion. Self-contained network operation is designed to protect and deliver messages from the source Host to the destination Host.

14 KEY WORDS	LINK A		LINK B		LINK C	
	ROLE	WT	ROLE	WT	ROLE	WT
Computers and Communi on						
Store and Forward Communication						
ARPA Computer Network						
Honeywell DDP-516						
IMP						
Program Design - IMP						